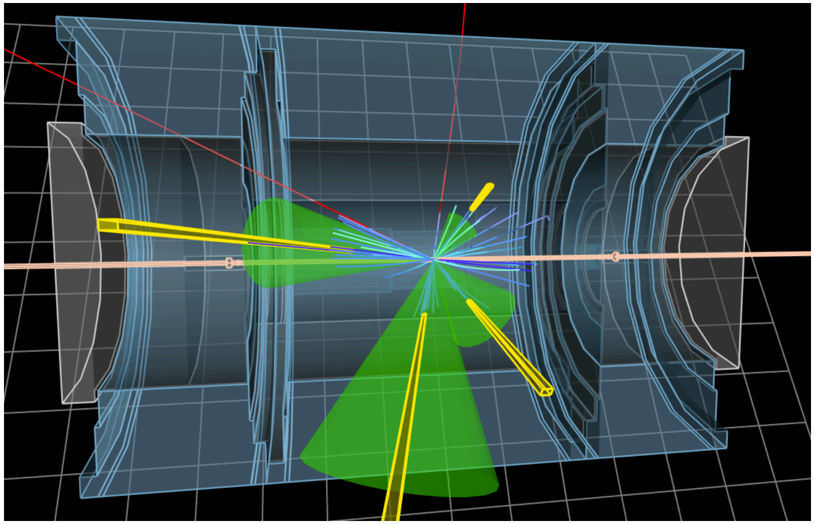
HEP Collider HPC Use, Prospects and Wishes



Tom LeCompte
High Energy Physics Division
Argonne National Laboratory

Outline

- Overview of the Science
- Overview of HPC use Today
- Some Extrapolations to the Future

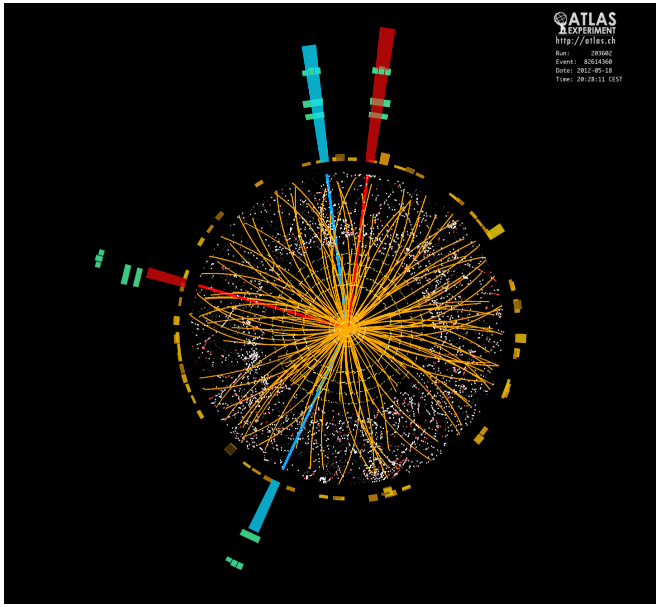
I am an LHC experimenter. This talk is arranged around that experience – at the present time, we are the only major experimental HEP group using HPCs at this scale.

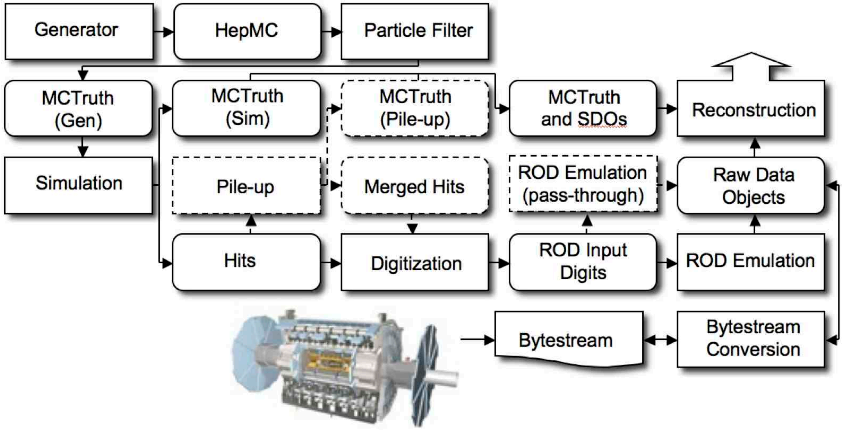
Collider Physics for Non-Physicists

- We collide particles together and measure the trajectories of the products in our detectors

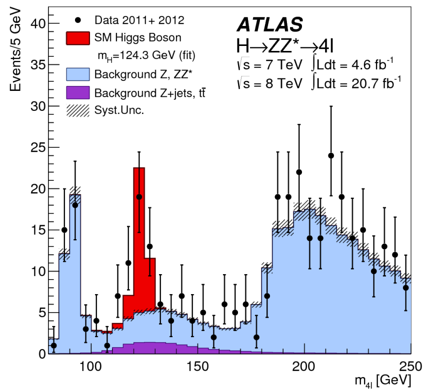
- We then compare these results with simulation – at multiple levels
  - Does the detector respond to these particles in the way we expect?
  - Do we see the number of particles in various categories that we expect?
  - Etc.

- This is a complex process, and *we are as dependent on the simulation chain* as we are on the data chain.
  - ATLAS uses about a billion cpu-hours per year on this.





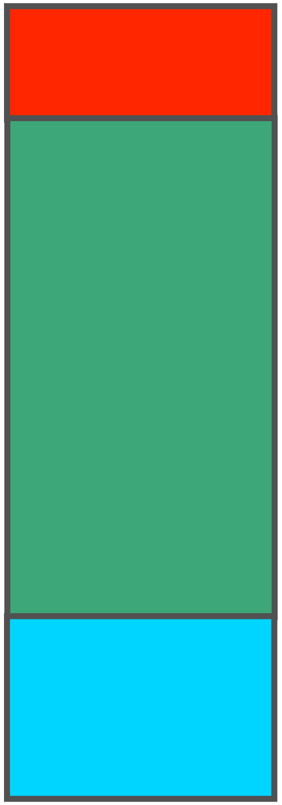
LHC Science Goals



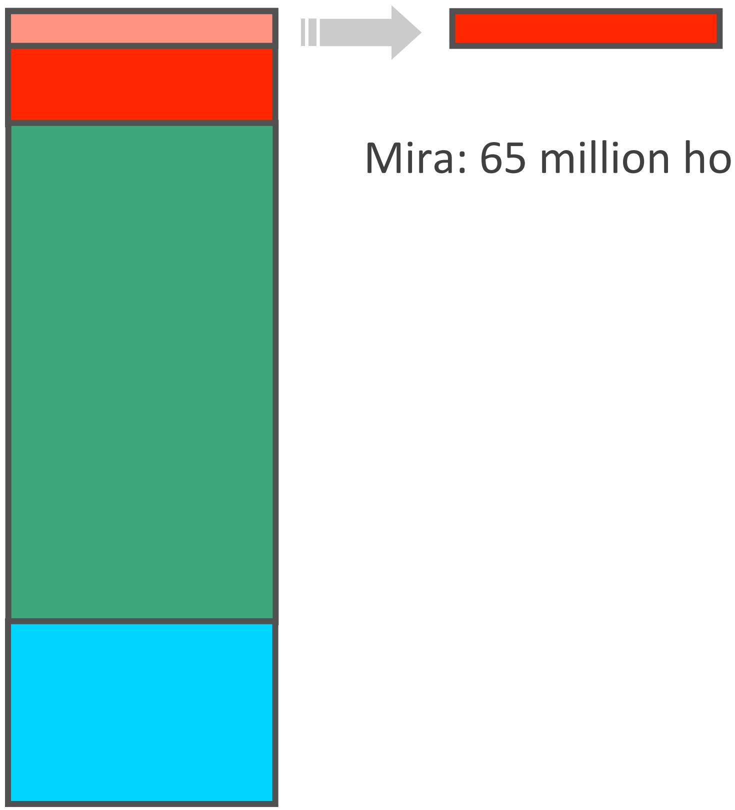
- Discovery of the Higgs Boson was nice.   But…
  - We don't understand why it's mass is what it is: it's too light to be heavy and too heavy to be light. This is suggestive of other particles lurking just beyond present sensitivity.
  - We don't know if this is the only Higgs boson, or if there are more.
- We don't know what Dark Matter is. But…
  - It sure would be nice to be able to make some in the laboratory.
- To answer these questions, we would like to collect ~100x the data at close to design energy (13-14 TeV, rather than the 7 & 8 TeV we have)

Computing to Reach The Science Goals

- ATLAS uses about a billion CPU-hours per year on the Grid
  - This does not include the cycles spent calibrating or reconstructing the data; the problem is defined as what happens after this point
- Event Generation
  - Simulate the physics process of interest: produces lists of particles and their momenta
- Simulation
  - Simulate the interaction of these particles with the detector
  - You may have heard the term "Geant" or "G4". Geant4 is the toolkit by which we do this.
- Reconstruction and Analysis
  - Treat the simulated data as real, reconstruct the particles, and do the final analysis



ALCC-2014 Use



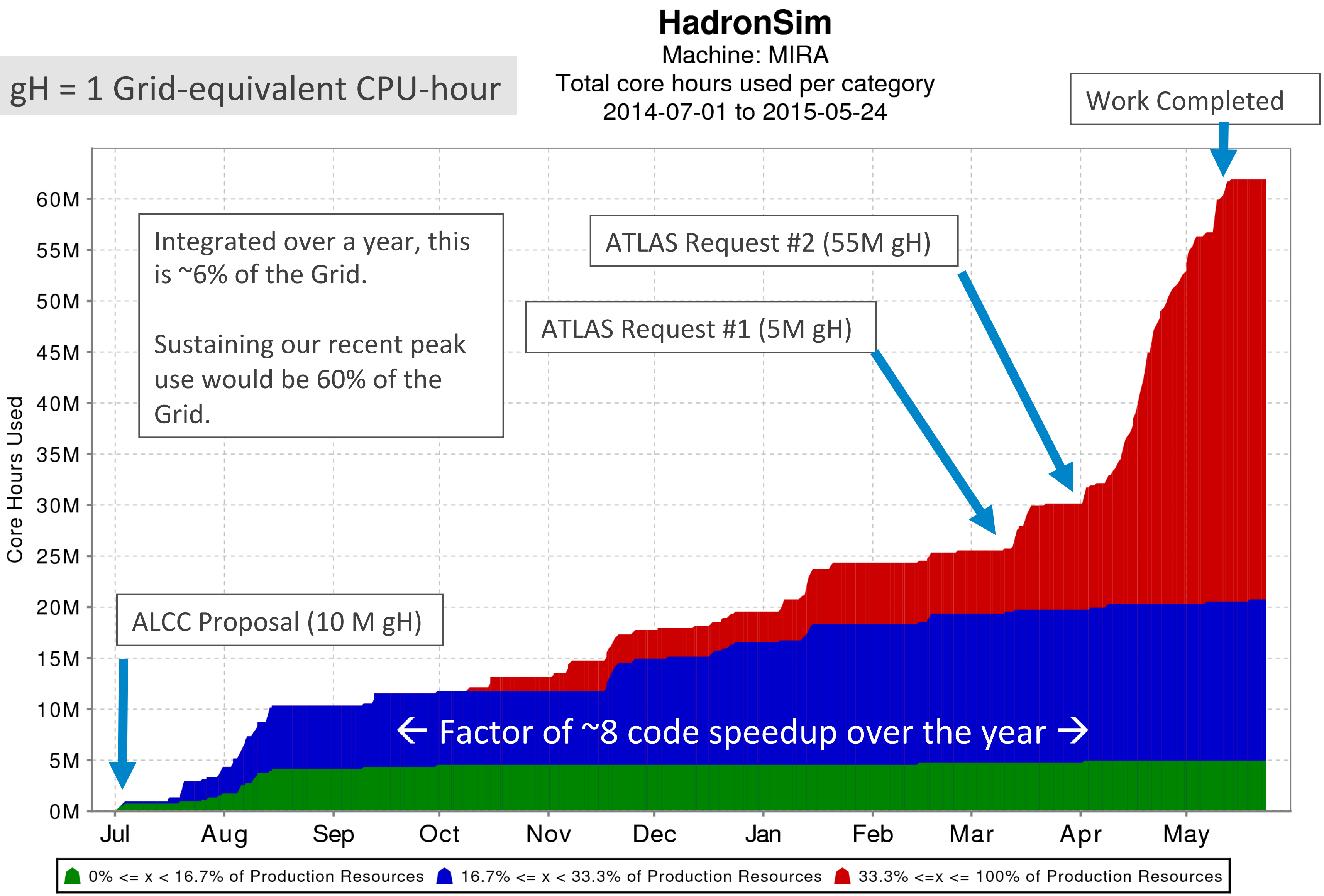
Mira: 65 million hours

Grid: One billion+ hours

- 70 million Grid-equivalent cpu-hours of event generation have been run on Mira and thereby offloaded from the Grid
  - 2 FTE worth of effort
  - If we were a country, we would be the 7th largest provider of cycles to ATLAS
- This is only event generation (red), but the recovered cycles can be used for whatever the experiment wants
  - This is dominated by one event generator (called Alpgen)

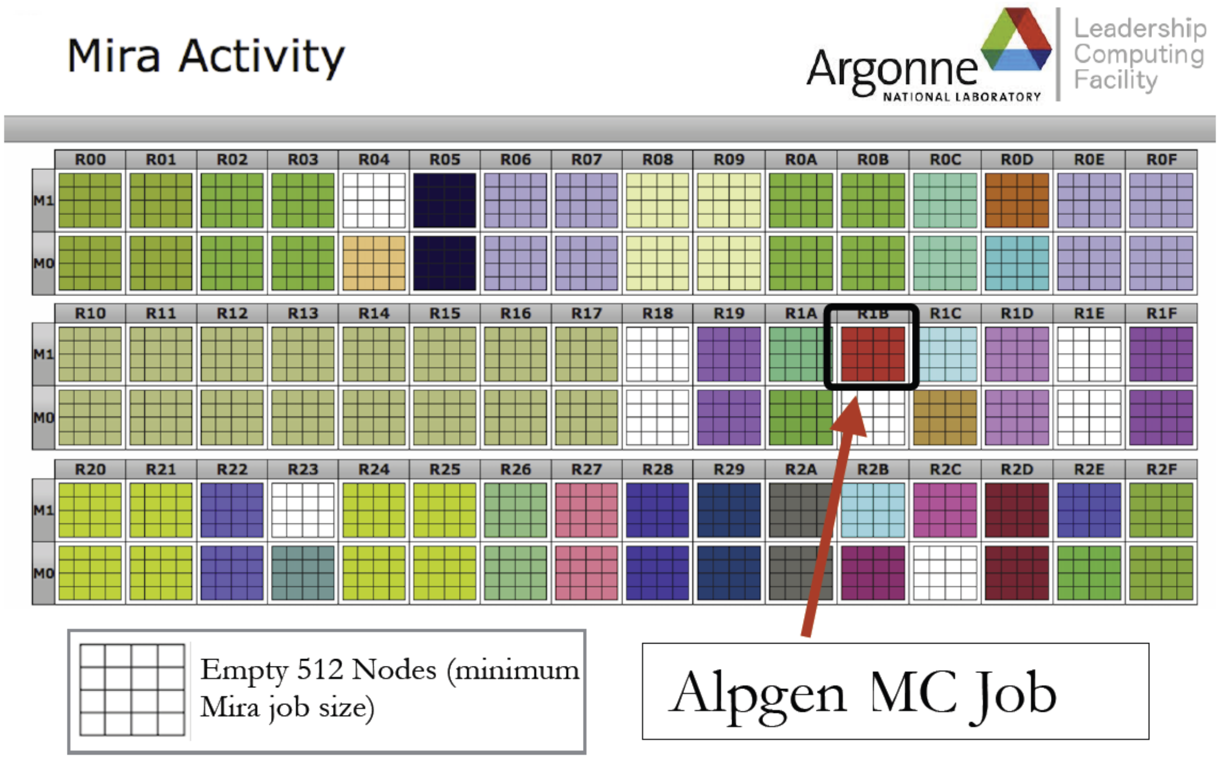
Mira ALCC Use - Visually

1 gH = 1 Grid-equivalent CPU-hour



Where we were 1 year ago

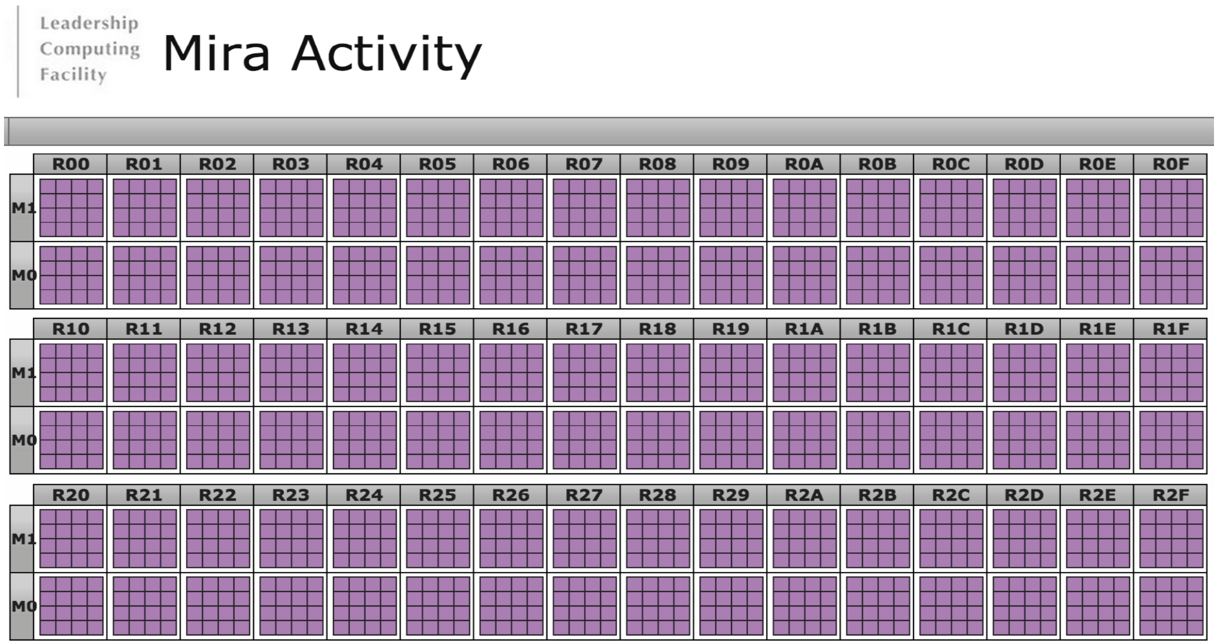
- We could run in the minimum Mira partition (and only in the minimum efficiently)
- Event generation rate was 1/15 of a Grid node
  - ALCF suggests a nominal of 1/10



At this time we were running, but limited by I/O.

Where we are today

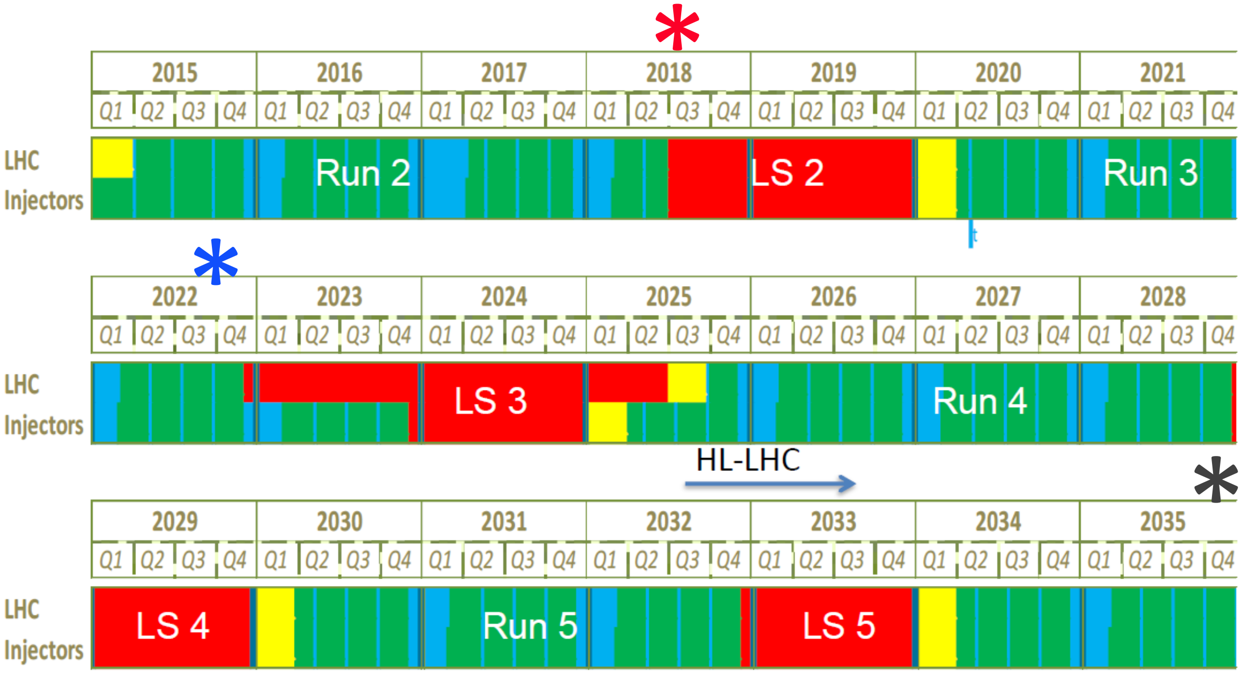
- We can run using the entire machine
  - For throughput reasons, we normally limit ourselves to 1/3 of the machine: a million parallel processes
- Per core event generation rate is ~1.5x a Grid core
  - Speedup of x23 over the year
  - Mira has a lot of cores! (768432)



While this job was running, Mira was producing the equivalent computing as 5 or 6 ATLAS Grids.

On our best days, we provide the equivalent computing capacity of the whole ATLAS Grid.

Future Plans

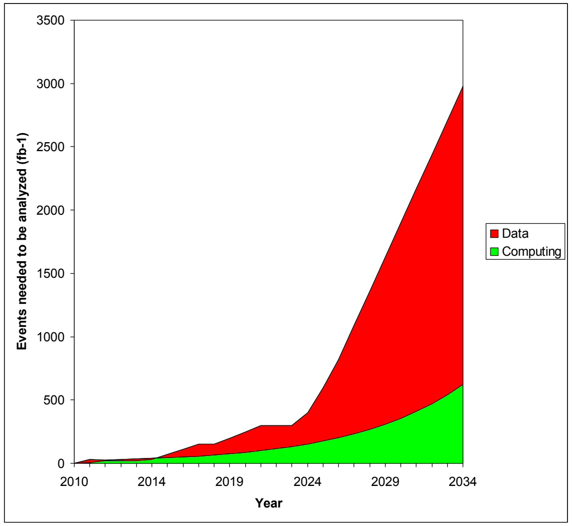


- * Have 3x today's data
- * Have 10x today's data
- * Have 100x today's data

High Performance Computing Motivation in HEP

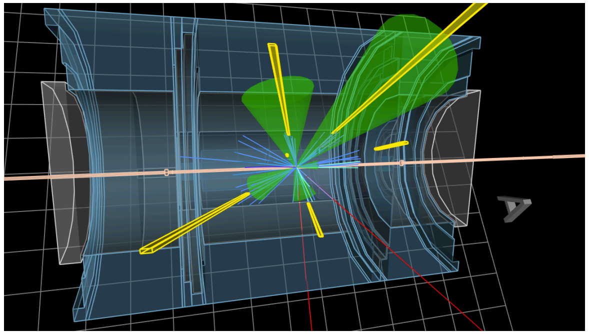
There are two problems we are trying to solve

- The capacity problem
  - Our needs are growing faster than the Grid is growing – and likely can grow



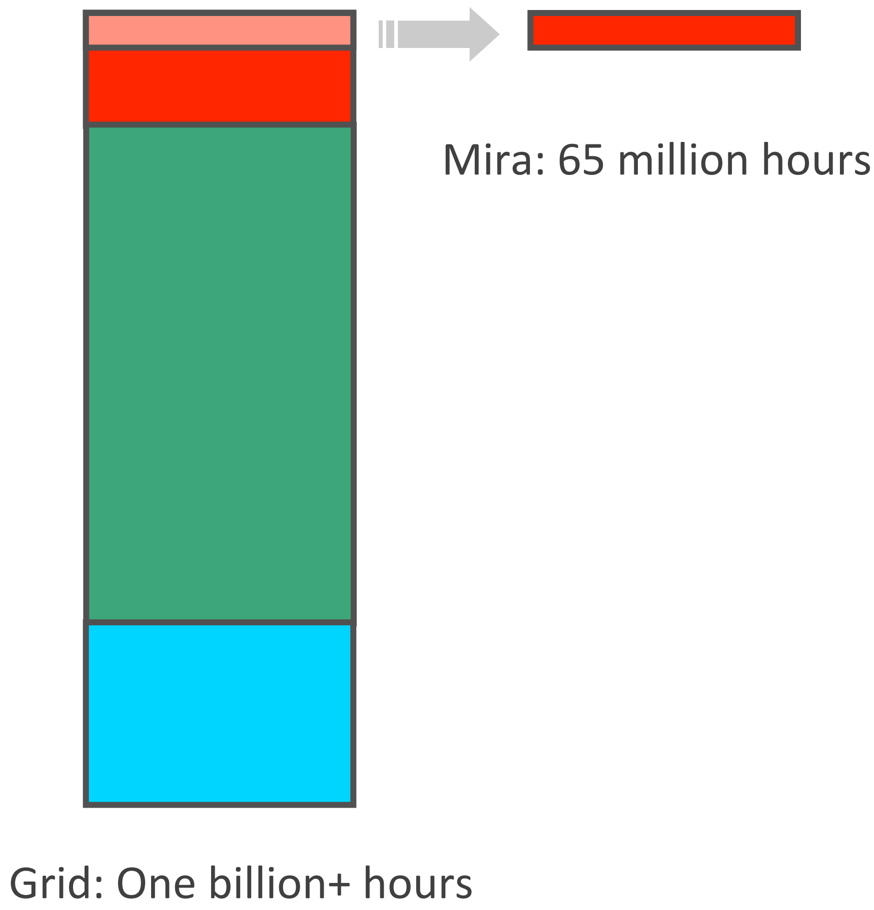
The green assumes 15% growth per year from Run 1, and that Run 1 had exactly enough capacity.

- The capability problem
  - Some problems can't run on the Grid
    - Highly filtered samples can produce zero events in a job – which looks like a failure and triggers a resubmit. Which triggers a resubmit. And so on.
    - Sherpa integration times stretch into the weeks, but can be done overnight with a few hundred threads (scaling is poor today, but this makes doing this *possible*. Efficient comes later)



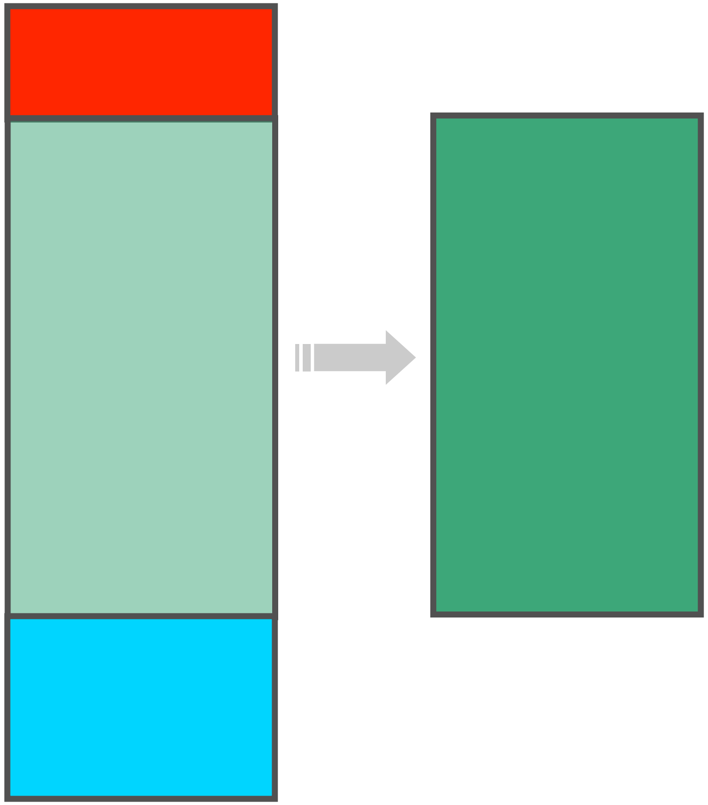
We see HPCs playing a role in both of these

Growth Area: Event Generators



- We could – and intend to - double this by porting a 2nd generator, Sherpa, to Mira.
- If we succeed in scaling the Sherpa integration phase, this could increase by an order of magnitude
  - We would be able to simulate events we couldn't simulate before. These events will be highly in demand.
- We would like the number of simulated events to scale with the number of real events
  - Without HPCs, this looks unlikely

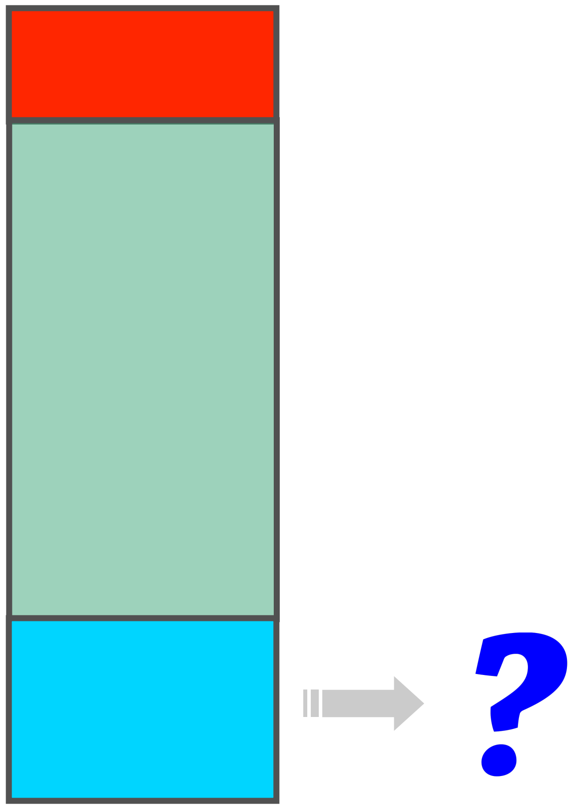
Growth Area: Simulation



- Virtually all simulation could be done on HPCs
  - The Geant4 kernel runs well today
  - The ATLAS software framework would need to be ported*
    - This is more a long problem than a hard problem.
- Unlike event generation, this is data-intensive
  - Today occupies 10's of PB
  - Scales with the data collected
    - Around 2025 will reach 1 EB.

* Runs on Edison today; but all announced supercomputers have different architectures.

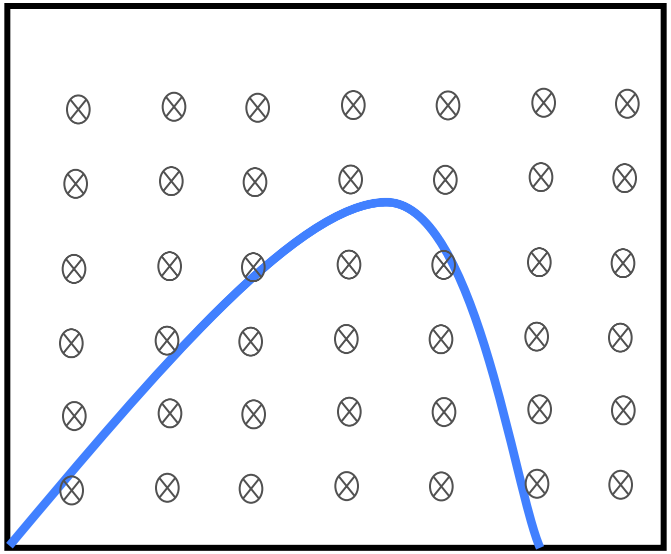
Growth Area: Reconstruction and Analysis



- Reconstruction is the use case the Grid was built around – this is I/O heavy
  - Under what, if any, circumstances does it make sense to move this piece to HPCs?
  - I don't know the answer today – but the it is probably related to data availability at the HPC sites

- Analysis might grow by a lot
  - The analysis paradigm is determined by the amount and nature of the computing available.

How We Set Limits Today



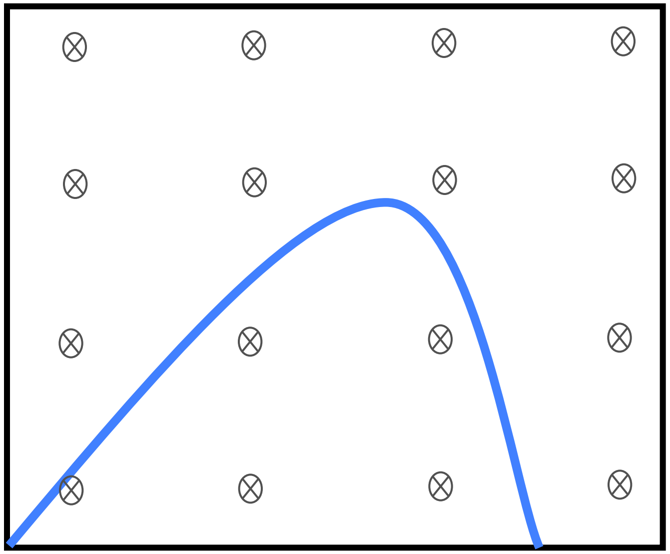
Suppose our theory has parameters x and y, e.g. masses of unknown particles. We then take a grid of (x,y) points, and see if the point is allowed or excluded by the data.

Based on the outcome of those tests, we interpolate an exclusion curve.

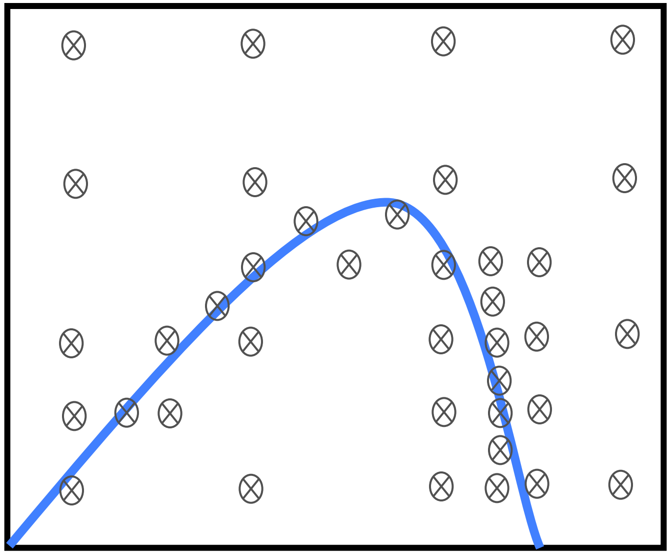
These points are run on the Grid, so run at different places and time. There is no opportunity for coordination.

How We Might Set Limits Tomorrow

Start with a coarse grid



Iteratively refine the grid



These points are run on the at the same place and time.
The program can coordinate, and converge on a better answer, faster.

Additional Benefits

- The past slide shows how better limits can be obtained at reduced computational cost.
- What if we were to increase computational cost?
  - We often have models with >2 parameters
  - We set limits in 2-d slices, with the other parameters kept constant
    - "Doing anything else would require a supercompter!"
  - Could we set 3-d limits?

2-d slices don't easily separate this

from this.

Summary

- In a year, ATLAS has grown from ε HPC use to 6% of the Grid.
- This could grow to 50% or more
  - One challenge is adapting the framework
  - Another is the data volume and mobility (it has to get back to the Grid)
- In addition, with the new LHC data our needs will grow
  - A factor of ~3 in 2-3 years
  - Another factor of ~3 in another 2-3 years
  - Ultimately a factor of ~100.
- Nevertheless, this is growing more slowly than the LCF capacity – **we can fit**

- It is likely that our needs will grow beyond this if the computing situation permits

- Other experiments are starting to show interest as well: perhaps another factor of three on top of this.